Multimodal Data Fusion As a Predictior of Missing Information in Social Networks

by

Jingxian Mao

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved November 2012 by the
Graduate Supervisory Committee:

Ross Maciejewski, Chair
Gerald Farin
Yalin Wang

ARIZONA STATE UNIVERSITY

December 2012

ABSTRACT

Over 2 billion people are using online social network services, such as Facebook, Twitter, Google+, LinkedIn, and Pinterest. Users update their status, post their photos, share their information, and chat with others in these social network sites every day; however, not everyone shares the same amount of information. This thesis explores methods of linking publicly available data sources as a means of extrapolating missing information of Facebook. An application named "Visual Friends Income Map" has been created on Facebook to collect social network data and explore geodemographic properties to link publicly available data, such as the US census data. Multiple predictors are implemented to link data sets and extrapolate missing information from Facebook with accurate predictions. The location based predictor matches Facebook users' locations with census data at the city level for income and demographic predictions. Age and relationship based predictors are created to improve the accuracy of the proposed location based predictor utilizing social network link information. In the case where a user does not share any location information on their Facebook profile, a kernel density estimation location predictor is created. This predictor utilizes publicly available telephone record information of all people with the same surname of this user in the US to create a likelihood distribution of the user's location. This is combined with the user's IP level information in order to narrow the probability estimation down to a local regional constraint.

TABLE OF CONTENTS

Page

LIST OF TABLES iv

LIST OF FIGURES v

CHAPTER 1

1 INTRODUCTION 1

2 SOCIAL NETWORKS 5

2.1 What are social networks? 5

2.1.1 Online Social Networks 6

2.1.2 Social Network Structure 8

2.2 Analysis in Social Networks 9

2.3 Social Network Visualization 14

2.4 Using Social Networks to Predict Missing Data 18

3 METHODOLOGY 21

3.1 Application on Faceboook 21

3.2 Data 24

3.2.1 Facebook Data 24

3.2.2 Census Data 27

3.2.3 Geographical Location Data 29

3.3 Data Analysis 31

3.4 Data Visualization 38

3.4.1 Overlays on Google Maps 38

3.4.2 Node-Link Network 42

4 RESULTS AND ANALYSIS 46

4.1 Case Study 1 46

4.2 Case Study 2 53

5 CONCLUSION AND FUTURE WORK DISCUSSION 57

Page
REFERENCE 60

LIST OF TABLES

TABLE Page

3.1 Table of Data set From Facebook 27

3.2 Census Data Table - State Level. 28

3.3 Census Data Table - County Level. 28

3.4 Census Data Table - City Level. 29

3.5 Census Data Table - City Block Level. 30

3.6 US city and county table. 30

3.7 US Census Geographical Location Table. 32

4.1 People's Incomes Table. 55

LIST OF FIGURES

FIGURE Page

2.1 Node-link diagram and adjacency matrix [36] 16

3.1 App Main Page 22

3.2 App Result Page 22

3.3 MySQL Tables. Tables with the prefix of "acs" contain census data; table "us_city_county" maps cities and counties into latitude and longitude; "us_names" maps names into geographical location . . 23

3.4 Requesting Permissions. If users agree to give out their permissions, this application will start to collect related data from Facebook. 26

3.5 Friends Map on Google Maps 39

3.6 Census Data Overlay - State Level: Zoom level from 1 to 4 41

3.7 Census Data Overlay - County Level: Zoom level over 5 41

3.8 Kernel Density Estimation Overlay 42

3.9 Node-Link Network 43

3.10 Filter Result 44

3.11 Brushing 45

4.1 Friends Google Map before the location prediction 47

4.2 Friends Network before the location prediction 47

4.3 Friends Google Map after prediction 48

4.4 Friends Network after prediction 48

4.5 KDE Prediction on All US States 50

4.6 KDE prediction on Specific States 50

4.7 KDE Prediction on All US States 50

4.8 KDE Prediction on All US States 51

4.9 KDE prediction on Specific States 51

FIGURE Page
4.10 Information Window 51
4.11 Information Window 52
4.12 Friends Google Map of "Jingxian Mao" 53
4.13 Information Window of People in Phoenix 54
4.14 Information Window of People in Phoenix 54

Chapter 1

INTRODUCTION

Different from the traditional content based information networks, online social networks not only allow users to gather information from websites, but to post and share information to others as well. Users of online social network sites can update statuses, post photos and videos, and share information with each other. However, not everyone shares all their personal information on these online networks. Furthermore, some information about users may be interesting in terms of marketing and geodemographics; however, such information may never be shared by the user (e.g., income).Yet, it is possible to approximate interesting details about a user by linking attributes collected from disparate data sources. Approximating data by linking social norms is a challenging task. A simple solution would be to send out surveys to the users as a means of collecting information. This solution is quite accurate; however, it is cost prohibitive and many users may not participate. This thesis explores means of extrapolating missing information in online social networks through an application of multimodal data fusion. The application links multiple data sets that are collected from social networks (e.g., Facebook) and other publicly available resources (e.g., CensusBureau) with each other. The missing information is then extrapolated through statistical analysis.

One problem to be solved in this thesis is linking population norm data (e.g., Census data, etc.) to social network data as a means of extrapolating information about a user. To do this, I have developed a predictive model for linking population norms to social networking data based on user provided data. From the social network data, I gather the Facebook data that contains users' basic information (e.g., age, relationship status, and location informa-

tion) and census data that includes yearly incomes of each state/county/city. A Facebook user is then linked to a single row in the census data table that has the same location with this user. Therefore, this user's yearly income can be predicted according to the yearly mean income of the state/county/city that the user is located in. Furthermore, with the information of this uesr's age and relationship status, I apply a linear equation to approximate the income prediction by combing two values in this row.

The goal of this thesis is to develop a scheme for creating a geodemographic profile of users within an online social network. This is done through the following two steps: (1) collect and link multiple data sets and (2) extrapolate missing information of social networks by analyzing the properties of these data sets. To achieve these goals, I have developed an application in Facebook to capture social network information. I then focus on predicting a very specific social norm, particularly income of the user as a proof of concept for this task. To do this, I attempt to capture the Facebook user's name, age, gender and location. However, not all of this information is publicly available. As such, two income predictors are created to link the Facebook data with public census data: a location based predictor, and an age and relationship based predictor. However, before applying the income predictors, location predictors are first established to predict the relative location of the user. The first predictor is the kernel density estimation location predictor. This predictor links the collected name data to the telephone directory data and outputs the distribution of people who share a surname with the application users. By analyzing the distribution, a user's location can then be predicted with a random threshold. The other location predictor is an IP address based predictor. Based on the user's IP address, this predictor locates the user at the census

tract level. Combining these two location predictors allows us to narrow down the location in which a user may live. Once the location is obtained, I can then link the age and gender details of a user with the census data of this location to determine social norms (such as income). Thus, with the data sets gathered from Facebook and publicly available resources, missing information of users in social networks can be predicted.

The rest of this thesis has been divided into four parts: (1) social networks; (2) methodologies; (3) results and analysis; (4) conclusion and further work discussion.

Chapter 2 gives a brief introduction about social networks and online social networks, including their definitions and brief histories. This chapter also introduces social network structures and several social network analysis methods. Furthermore, this chapter discusses social network visualization with a brief history and several network visualization tools.

Chapter 3 describes the primary contribution of this thesis and is divided into four sections: (1) Application on Facebook; (2) Data; (3) Data Analysis and (4) Data Visualization. Section 3.1 introduces the application on Facebook. Section 3.2 discusses what kinds of data are needed and how to collect the data. Section 3.3 discusses the data analysis. This section also consists of discussions about predictors applied in this application. Section 3.4 discusses data visualization methods that are applied in this application. The first part of this section describes the three overlays that have been rendered and the purpose of rendering these overlays. The second part introduces a node-link network that shows the relationships between the user's friends and this person. It provides several interactions for users to brush and filter.

Chapter 4 discusses some results of running this application on Facebook. It also discusses some aspects that affect the accuracy of these results. Chapter 5 concludes the work I have done within this thesis and discusses some problems that are not solved in this thesis.

Chapter 2

SOCIAL NETWORKS

2.1 What are social networks?

A social network is a structure that is made up of individuals or groups of people with some pattern of interactions or "ties" between the individuals or groups[47]. Two centuries ago, French sociologist Émile Durkheim and German sociologist Ferdinand Tőnnies introduced the concept of social networks known as social groups [12]. The term "social network" was first used by Georg Simmel when he explored the nature of networks [12] centuries ago. Since then, the topic of social networks has received much attention, not only in sociology, but also in fields such as psychology, anthropology, mathematics and computer science.

One of the key topics of exploration is the small-world problem. Milgram [45] concluded that in a small-world network, only 5 intermediaries (on average) are needed to connect two strangers in America. He also proposed a theorem based on the small-world problem which states that if two individuals cannot contact each other, there is also no way for the groups which they are embedded in to make contact with each other. In other words, these two groups are completely isolated from each other. According to Milgram's research, every large social network can be divided into small ones whose members are tightly tied together. These members usually share common information and have tight correlations between each other, and different groups are connected through common individuals. To discover correlations within and between social groups, Granovetter [31] explored the concept of 'strong' and 'weak' ties in social networks. In his model, he treats strong ties as connections within clusters while weak ties are connections between clusters. Strong ties group

similar nodes together and weak ties connect groups to form larger networks. Inspired by the small-world theory, Adamic [2] published a paper to discuss how to define clusters and study the connections within and between clusters in another man-made network - the World Wide Web. In this paper, he developed a search engine that can take the advantage of small world networks to show how it can influence advertising strategy. The result shows that analyzing social networks give us a clear understanding of a group's common properties and helps analysts to extrapolate missing information of a group.

2.1.1 Online Social Networks

An online social network is a social network based on online services and platforms. Boyd and Ellison [5] define online social networks as web-based services in which individuals are: (1) allowed to create a public or semi-public personal social web-page with a unique social link; (2) allowed to share and exchange connections and interesting information with other individuals; and, (3) allowed to view and traverse their connections and social activities, such as statuses, profiles, pictures and videos within the services.

The first recognizable social network site (SNS) Classmates.com[14] was created by Randy Conrad in 1995. In comparison to many current social network sites (e.g., Facebook, Renren and Google+), Classmates.com is much simpler. It is focused on helping users to find, connect and keep in touch with their old friends and colleagues from school. In 1997, another social network site named Sixdegrees.com[1] emerged. It is named after the theory that two strangers in the US can get in touch with each other through 5 intermediaries [45]. It expanded the work in Classmates.com by allowing family members and friends from other groups to join and interact.

In the 2000s, SNSs came to further prominence with the development of Ryze.com, which was launched in 2001. Ryze.com is the first SNS built for business use. The success of Ryze.com signaled that online social networks were no longer only for people to get in touch with their old friends, they could help people network for jobs and expand businesses as well. A year later, SNSs expanded to other fields. Friendster is one of the notable SNSs that was designed to compete with Match.com, a prominent online dating site. Instead of creating opportunities to meet complete strangers, Friendster helps people who have mutual friends meet each other. The founders thought it was more likely that people with mutual friends would be able to form long lasting relationships. In 2003, more general-purposed SNSs were launched. In the field of professional business, there was LinkedIn, Visible Path, and Xing. MyChurch joins Christian churches and their members. Couchsurfing connects travelers to people with couches [5]. Among these new SNSs, MySpace was one of the most significant and popular sites. In MySpace, users are allowed to customize their own profiles and add specific features to their social web-pages. In 2006, Facebook was released. Since then, Facebook has become one of the most popular SNSs in the world. Facebook's success not only attracted attention in the US, it also led to a global SNS wave. In Boyd and Ellison's paper [5], we know that, "Orkut became the premier SNS in Brazil before growing rapidly in India (Madhavan, 2007), Mixi attained widespread adoption in Japan, LunarStorm took off in Sweden, Dutch users embraced Hyves, Grono captured Poland, Hi5 was adopted in smaller countries in Latin America, South America, and Europe, and Bebo became very popular in the United Kingdom, New Zealand, and Australia. Additionally, previously popular communication and community services began implementing SNS features. The Chinese QQ instant messaging service instantly became the largest SNS worldwide when it

added profiles and made friends visible (McLeod, 2006), while the forum tool Cyworld cornered the Korean market by introducing homepages and buddies (Ewers, 2006)."

2.1.2 Social Network Structure

A social network is constructed by three components: users, groups and relationships.

Users

Users are the essential component of social networks. A social network is formed by groups of separated individuals. For each social network site, individuals must register an account within that network. Once users have registered in an SNS, a personal page, which is called profile page, will be created and a unique identity will be assigned within that SNS. Users can edit their information (e.g., birthday, home address, telephone number, interest, etc.), change the background and layouts of their profile page, change their profile pictures, update their personal status on the SNS and decide who has the authority to retrieve their personal information.

Group

People usually are not alone in the SNS. There are groups formed by individuals who have common interests. For example, a group of people connected in a social network may all like tennis. It is very likely that they know each other and play or watch tennis together. As a result, they construct a tennis group. People who graduated from the same college would probably form a group only for the alumni. In a social network site, users can create a group by themselves. Facebook allows users to create almost all kinds of groups without any limitation. Users can add or remove people from groups. Once users are

members of a group, they can post onto the group page and communicate with other members.

Relationships

Relationships, or links, in an SNS represents the connections between users. Typically, there are two possible connections. If two people know each other, then they are directly linked; if one knows the other while the other does not know the one, then they are not connected. The connection is represented by an undirected link for the first case while the connection is represented by a directed link for the latter one in Graph Theory. Some SNSs (e.g., Google+ and Facebook) only allow users to link with each other when both sides of links agree and form an undirected link. Some SNSs (e.g., Twitter) allow users to follow whoever he or she wants to in order to form a directed link to show users' interests. Users link to others for various reasons. Some people tend to connect with professionals, link with acquaintance or connect to users with common interests. Others want to follow celebrities and get to know their recent activities. Since most of SNSs allow users to browse others' profiles and links, people can browse people's information by retrieving user-to-user links through SNSs.

2.2 Analysis in Social Networks

With the fast growth in popularity of social network sites, social network analysis (SNA) and visualization have attracted attention from professionals in sociology and information visualization. Social network analysis analyzes structure of social networks and relations between individuals. Gretzel [32] concluded that there are 4 important concepts: "(1) Individuals and their actions are treated as interdependent rather than independent, autonomous units; (2) relational ties (linkages) between actors are channels for transfer

or "flow" of resources (either material or nonmaterial); (3) network models focusing on individuals view the network structural environment as providing opportunities for or constraints on individual action; (4) network models conceptualize structure (social, economic, political, and so forth) as lasting patterns of relations among actors."

Even though there are still other analysis methods for social networks, visual and mathematical analysis are the most popular and significant ones. In visual and mathematical analysis, individuals or actors are represented by nodes while connections between people are represented by edges. Wasserman, S. and K. Faust [62] divide the analysis of network data into 5 levels:

- actor level - centrality, prestige and roles
- dyadic level - distance and reachability, structural and other notions of equivalence
- triadic level - balance and transitivity
- subset level - cliques, cohesive subgroups, components
- network level - connectedness, diameter, centralization, density

In these 5 levels, it is of utmost importance to decide the location of each node (actor level) in a network. With these measures of actors' locations, developers and analysts can develop insight into the various roles and sub-networks in a network, like who are the connectors, centers, bridges and isolates; what are the clusters and which nodes are within the clusters. One means to measure the location of each node is to calculate the node's centrality – Degree Centrality, Betweenness Centrality, and Closeness Centrality.

- Degree Centrality

Degree centrality is decided by a node's degree which shows the number of direct connections a node has. The degree can represent the immediate flows of a node in a network. By computing the degree of every node in a network, one can determine the "connector" or "hub" of this network. It is commonly considered the most significant aspect in a personal network.

- Betweenness Centrality

Betweenness centrality measures a vertex within a graph. It is a measurement that displays the importance of a node with respect to the communication between other nodes in a network. Usually, the node with a high betweenness centrality in a network plays the "broker" role. It is a single point of failure of connections of the whole network and has great influence over the flows in a network.

- Closeness Centrality

Closeness Centrality is a measurement for finding the length of the shortest path of each node. This centrality decides which node is closer to the center of a network. With a lower total distance to all other nodes, a node is more central in a network.

The degree centrality, betweenness centrality and closeness centrality provide insight of individuals' location in a network. However, it is not enough for an overall network analysis. An analysis on the relationship between centralities of all nodes in a network is needed to reveal the overall network structure. If a network is highly centralized, it must be dominated by one or a

few central nodes. Disabling (e.g., removing or damaging) these central nodes easily leads to a failure of the network. On the other hand, if a network is less centralized, there is no central node, or there are so many central nodes that they barely have influence on the others. Disabling one single central node can make several related nodes fail but other parts of the network may still work well. In order to design a better network structure, a developer also needs to consider other aspects, such as network reaches, network integration, boundary spanners, etc.

As mentioned in the beginning of this chapter, individuals' properties and their connections with others in social networks can influence one's understanding of social structure, advertising strategies, policy planning, etc. Therefore, researchers are very interested in online social networks analysis to explore the information behind data in social networks. Papacharissi compared three online social networks (e.g., open-to-all Facebook, professionally oriented LinkedIn and the members-only ASmallWorld) to analyze the underlying structures and architectures of these three sites[50]. He found that Facebook has the most flexible architecture with a publicly open structure, looser behavioral norms and free tools for users to connect with others, while LinkedIn and ASmallWorld are tighter in that they provide less open space for users. To more clearly understand which online social networks features attract users, Schneider et al.[56] studied users' interaction with four online social networks - Facebook, LinkedIn, Hi5, and StudiVZ. As a result, they found that users are more likely to keep working on the same activities (e.g., photos, and messaging). They also noticed that for SNSs like Facebook, users can be on these SNSs for a long time but they have little interactions with SNSs. However, the users of the professional SNSs, like LinkedIn, are totally

different. They tend to stay within the SNSs only when they feel like interacting with SNSs. Liben-Nowell [43] surveyed several mathematical models of social networks dealing with the small-world phenomenon to find efficient routes that users can reach individuals in a large-scale social network. Caverlee and Webb [8] studied the characteristics of large online social networks including the sociability of users, demographic characteristics, and text artifacts of MySpace users through a large-scale of MySpace. In their discoveries, nearly half of profiles on MySpace have been abandoned. Only accounts with vast majority of friends, comments and group activities are still activate. They also found that the patterns of language use for users and popularity of users on MySpace are influenced by the users' age, location, and gender.

To study local usage of online social networks, Yardi et al.[63] analyzed two geographically local events - a shooting and a building collapse in Wichita, Kansas and Atlanta to see how quickly people respond to local accidents in Twitter - an online social network. In their research, they find that local networks are denser than the non-local network and central individuals in the Twitter network are also located in the real world. Takhteyev et al.[58] investigated correlations and formation of social ties on Twitter. In this paper, they discover that (1) ties that lie within the same metropolitan region have common interests and posts; (2) distance, borders, and language differences are the three aspects that most impact non-local Twitter links; (3) the best predictor of ties are airline flights. Onnela et al.[49] believe that geography is always constraining our social interactions because social groups are fundamental building blocks of our societies. They investigated these constraints according to a user's geographic location. They pointed out that small groups are geographically tight while large groups are much sparser. Cho et al. [11]

discussed the correlations between human movement and mobility within social networks. They find that long-distance travel is more influenced by social network ties while short-range movement is less correlated to these ties.

2.3 Social Network Visualization

Visualization provide investigators with tools to help them explore the network and gain insight into the network structure. To make a good graph visualization, a simple problem: "given a set of nodes with a set of edges (relations), calculate the position of the nodes and the curve to be drawn for each edge."[37] needs to be solved.

In Freeman's paper [24], he introduces the use of pictorial images in social network analysis and reviews the long history of visualization applied to social network analysis. The first time that visualization was used in social network analysis can be traced back to the 1930s when the visualization graphs were produced by hands. In the early 1930s, Moreno introduced the node-link diagram in his work and generalized several ways to draw such a diagram. In his graphical representation work, he used directed relation links to show the response from one actor to another. He also introduced the idea of representing different classes or groups by using different shapes or colors of symbols. He also used geographical location to indicate the relations among nodes, which is still an important aspect for visualization today.

In the 1950s, scholars realized that Moreno's approaches did not provide a good method for finding the location of each node in a network. The way to position a node is according to sociometric status (e.g., degree, centrality). To make the locations of nodes more accurate, Proctor [52] came up with an idea to use computational procedures to aid in placing points. They used factor analysis to partition actors into several separate groups and calculated

the combination of variables for finding the two best ranked combinations of variables. Then they positioned points onto a 2D space with these two variables being the axes. In this way, close points usually have similar attributes. Later in 1966, Laumann and Guttman [41] introduced another method called multidimentional scaling (MDS) to locate points from nD space to mD space where n > m. MDS breaks the restriction of factor analysis that the latter can only project nD space into 2D space. It also shows the relations among points by putting similar points close to each other. In 1979, correspondence analysis was introduced by Levine [42] to locate nodes to the centroids of the points that they have chosen or the ones that can represent the whole network.

The most popular forms of graph visualization are node-link diagrams and matrices. In node-link diagrams, nodes represent the actors (people/users) in a social network and links represent the connections among the actors. For matrices, rows and columns represent individuals while the cells represent connections between two individuals. Keller et al. [40] show that the adjacency matrices are restricted to the engineering community for a variety of applications (e.g., process modeling, change prediction), while the node-link diagram has a more wide spread usage in the real-world. Since the adjacency matrices and node-link diagrams are very different in visualizing graphs and networks, Henry et al. [36] presented a hybrid representation to show the global structure of networks for the detailed analysis of local communities. They use the node-link diagram to show the global structure of a network while using adjacency matrices to support the analysis of communities (Shown in Figure 2.1).

There are other graph drawing algorithms besides the node-link diagram and adjacency matrices. Among these algorithms, most graph layouts

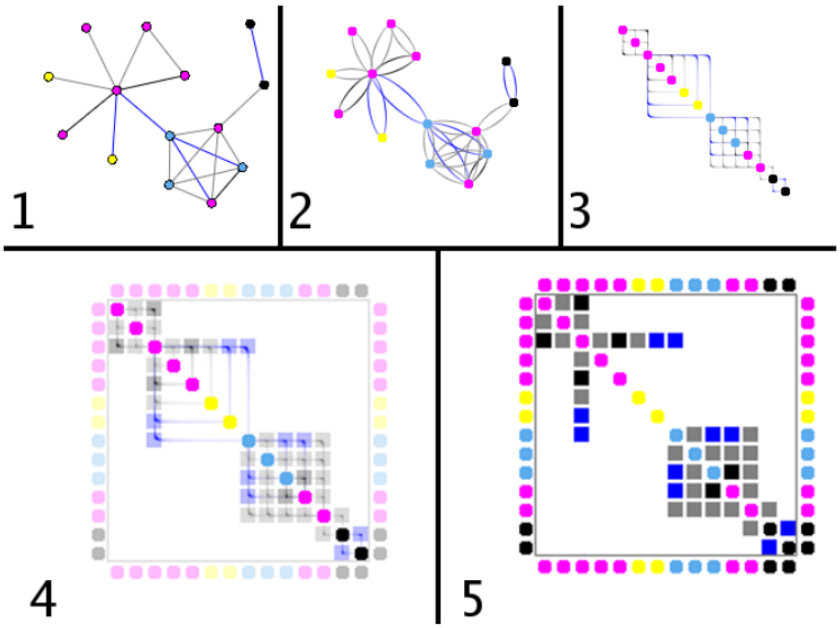

Figure 2.1: Node-link diagram and adjacency matrix [36]

are tree layouts. The classic tree layout generates a top-down layout whose children nodes are below their common ancestor. Reingold and Tilford[53] propose one of the best known layout algorithms (Reingold-Tilford algorithm) which can produce top-down and left-right tree layout. This algorithm clearly encodes depth level, avoids edge crossings, and draws isomorphic subtrees identically. In this algorithm, ordering and symmetry are preserved to create a compact layout. It starts with a bottom-up pass of the tree, and then merges left and right subtrees. Finally, the algorithm applies a top-down pass for assignment of final positions of nodes. Eades [16] introduced the radial layout in his book. This algorithm places nodes on concentric circles according to their depth in the tree. The root of the tree is in the center of the graph. If the depth of the node is deeper, the radius from the root to the node is bigger. The cone tree algorithm [7] generates a tree layout - "balloon view" in which the children nodes form a circle around their parents. Another graph layout

algorithm that needs to be mentioned is the Spring Layout, which is also called the Force-Directed Method. This method is proposed by Eades [17]. In this approach, nodes and edges of a graph are modeled as physical bodies tied with springs based on Hooke's law. Sometimes, data sets are too big to be visualized in a 2-D graph layout. Therefore, 3-D graph layout algorithms have been proposed in order to have more space to fill when uses data sets. A 3-D graph layout algorithm is the cone tree [54]. This method places nodes at the apex of a cone with its children evenly along its base. Later on, the cone tree algorithm has been revisited and improved by several analysts. Besides these traditional tree layouts, there are also other tree layouts. One of the most famous nontraditional tree layouts is tree-maps [38]. This algorithm generates sequences of nested boxes to represent a tree. Each box represents a leaf node and the biggest box represents the tree root. A box containing another box indicates that the enclosed box is a child of the box which encloses it. In tree-maps, the size of the box is significant, i.e. the size of the box is mapped to a specific attribute value of one node. For example, in a file system, a larger box represents a larger size of file or memory.

To better understand the social network structures and to make use of the data on online social network sites, many visualization tools have been developed for the online social networks. Heer et al.[34] developed a visualization system that analyzes online social networking services. Based on node-link network layouts, Vizster provides users with several visualization tools to analyze an online community and explore connectivity in large graph structures. Paul Mutton et al. [46] developed an online chatting system - Internet Relay Chat which allows groups of people to chat around the world. In this system, an IRC bot is used to monitor a channel and produce visualization methods on

the created online social networks. It reveals the structure of the online social networks, highlights and strengthens the connectivity between users, as well as clusters users into different groups based on their relationships. However, the graph is not the only method to visualize the social networks. Viegas et al. [60] discussed whether visualization of social networks can come in other forms besides graphs. They make a comparison between a traditional network graph with email contacts as nodes and a visualization method that depicts the temporal rhythms of interactions by applying depiction text and correlated images. It turns out that when the social network is too sophisticated, the graph based visualization can be cluttered and rather illegible. It is very necessary to discover other good visualization tools. Henry et al. [35] suggested that the locally density of social networks makes node-link diagrams unreadable. Thus they develop MatLink, which is an enhanced matrix visualization of graphs that overlays a linear node-link representation on the matrix. From their case studies, MatLink can produce an accurate result for a path-related task which is difficult for ordinary matrix visualization tools. This is done by adding dynamic feedback of path-relationship between nodes when the pointer moves.

2.4 Using Social Networks to Predict Missing Data

In most online social networks services, there are some users who do not provide their personal information.

An easy method to collect the missing data is to make surveys. However, researchers have tried to predict the missing data from the existing data on social networks and from other publically accessible resources. DeScioli et al. [15] noticed that human behavior has changed since the introduction of online social networks. They studied a case in the MySpace social network

that allows users to rank their friends. They found that human friendship is caused by cognitive systems which means people tend to trust strangers who are in their friends' top ranked list. Therefore, they believe that a prediction about people's best friends can be based upon how their partners rank that individual. Liben-Nowell et al. [44] discovered a formal model for geographic social networks to show the relations between geographic location and friendship. They introduce the notion of rank-based friendship which shows the correlation. This approach can find a user's friend's geographical location through friendship links, which helps analysts to guess an individual's geographical location.

Some researchers use IP-based algorithms to predict users' geographical locations [39, 19, 33]. However, IP-based geolocation algorithms are not very accurate. Poese et al. published a paper [51] that explores the reliability of IP geographical location databases. They compare several geographical location databases to investigate the limitations of these databases - (1) most data saved in the IP geographical location databases refer only to several popular countries and areas; (2) IP addresses changes too often which may not reflect the original allocation of IP blocks. Gill et al. [26] proposed a new way to handle both the delay-based IP geographical location techniques and more advanced topology-aware techniques. They tried to overcome the forged results that come from an adversary to make the IP geolocation more accurate. Wang et al. [61] developed a method to automatically analyze Web-based IP information to overcome the shortcomings of IP geolocation algorithm that most of IP addresses can only geolocate to the city level. Instead, Wang's method tries to bring it down to street-level. Backstorm et al. [3] developed an algorithm to predict the user geographical location more accurately based

on an IP geoloaction algorithm, which helps them to find correlations between geographical location and friendship.

Besides the IP goeloaction algorithms, researchers also study some other methods to predict user's geolocation. Cheng et al.[10] proposed and evaluated a probabilistic framework for estimating a Twitter user's city-level location based on the content of user's tweets. In their approach, there are three key features: (1) tweet contents that are posted by Twitter users; (2) automatic identifier for content words with a strong local geo-scope; (3) a smoothing model for local geolocation estimating. This method gives researchers, advertisers, and analysts more flexibility to find out what Twitter users want. It also shows where the Twitter users are. However, the accuracy of the location prediction is 51% for the users.

In this thesis, four predictors were created to approximate Facebook users' incomes and geographical locations, introduced in Chapter 3.

Chapter 3

METHODOLOGY

As one of the most popular online social network services around the world, Facebook recently reports that its active users have crossed the 1 billion mark. [59] In this thesis, I will explore ways to link social norm data (i.e., demographic/census data) to a user's social network data. This chapter introduces an application developed in Facebook which collects user's social network information and does multimodal data fusion to predict and link user's demographic data.

3.1 Application on Faceboook

Facebook provides several free application programming interfaces (APIs) for every developer to create applications in Facebook. Developers can choose their favorite languages (Javascript, Object C, Java, PHP) and platforms (websites, mobile) in which to interact with Facebook. In order to gather social network data, I have created a Facebook application that determines "How Posh" one's social network is based on a user's first degree social network links. My application consists of two main pages, the application instruction page and the data visualization tool which allows a user to explore the "poshness" of their network. As shown in Figure 3.1, the main page consists of the introduction and instructions of the application. At the bottom of this page, there is a button - "View Your Map" that redirects to the results page. Once the web page is redirected to the results page, a Google Map, a Node-Link network, an information window and a text result will be automatically displayed on this web page (Shown in Figure 3.2). The text result will display user's predicted yearly income and the rankings of this user's incomes among the US population and user's friends. On the Google Map, several layers have been

Figure 3.1: App Main Page

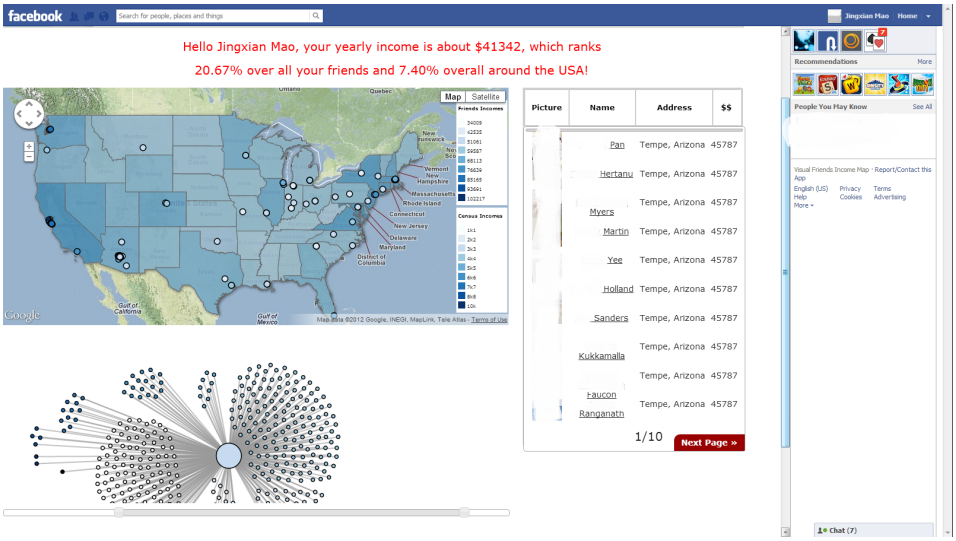

Figure 3.2: App Result Page

created. The blue nodes represent friends' geographical locations and predicted yearly incomes. The blue blocks display the demographics of people's average yearly incomes for each state and county. In the Node-Link network, each node represents a Facebook user. A link will be created if two people are friends on Facebook. All the nodes are clustered into 10 equal interval

bins based on users' incomes. When a user hovers the cursor over the nodes in either the Google Map or Node-Link network, the information window will display the user's profile information that is associated with the selected node.

	Table	Action
☐	acs_10_1yr	Browse Structure Search Insert Empty Drop
☐	acs_10_1yr_city	Browse Structure Search Insert Empty Drop
☐	acs_10_1yr_state	Browse Structure Search Insert Empty Drop
☐	acs_10_3yr	Browse Structure Search Insert Empty Drop
☐	acs_10_3yr_city	Browse Structure Search Insert Empty Drop
☐	acs_10_3yr_state	Browse Structure Search Insert Empty Drop
☐	acs_10_5yr	Browse Structure Search Insert Empty Drop
☐	acs_10_5yr_city	Browse Structure Search Insert Empty Drop
☐	acs_10_5yr_state	Browse Structure Search Insert Empty Drop
☐	acs_10_5yr_tract	Browse Structure Search Insert Empty Drop
☐	us_city_county	Browse Structure Search Insert Empty Drop
☐	us_names	Browse Structure Search Insert Empty Drop
	12 tables	Sum

Figure 3.3: MySQL Tables. Tables with the prefix of "acs" contain census data; table "us_city_county" maps cities and counties into latitude and longitude; "us_names" maps names into geographical location

On the server side, three web pages in PHP are created to interact with a MySQL database to query yearly incomes, city or county geographical locations and people's geographical locations in the US. The tables shown in Figure 3.3 contain different data sets. When a user clicks the "View My Map" button on the main page, the client side will send out three GET methods to the server. One is to query the city or county's latitude and longitude data in table "acs_10_5yr_city"; another one is to query the city or county's yearly incomes data in table "us_city_county"; and the last one is to query people's geographical locations according to their surname in table "us_names". After the server sends back the data that is asked for, this application will start to render the Google Maps, the Node-Link network and output yearly incomes.

3.2 Data

Data mining and machine learning are widely used in data analysis. How to collect, structure, and analyze data becomes an essential part of most analysis. In my work, I collect data from Facebook, Census Bureau [6] and the US Phone Directory.

3.2.1 Facebook Data

Facebook is one of the most popular online social network services. As previously mentioned, Facebook provides several APIs and SDKs for developers to download the data they are interested in from Facebook and to help them interact with Facebook. Facebook provides three APIs to let developers retrieve data from its database: (1) Graph API[21]; (2) Facebook Query Language (FQL)[20]; (3) Legacy REST[22];

- Graph API

The social graph is the core of Facebook. "The social graph in the Internet context is a sociogram, a graph that depicts personal relations of Internet users." [9] Every user on Facebook is a social graph object. Each social graph object has a unique id that enables developers to access its properties by requesting "http://graph.facebook.com/ID" from Graph API. The Graph API also presents views that represent the correlations between social graph objects [21].

- Facebook Query Language (FQL)

The Facebook Query Language is an SQL-style querying language that enables developers to query data from the Facebook Graph API. It also provides devel-

opers the ability of batching multiple queries in a single call. An example can be "GET $/fql?q = SELECT + uid2 + FROM + friend + WHERE + uid1 = me()$" which represents the query "SELECT uid2 FROM friend WHERE uid1=me()" [20].

- Old REST API

The old REST API also enables developers to interact with Facebook and query data from Facebook. It supports both OAuth 2.0 and lower authentication methods. Facebook is now trying to deprecate the REST API. As such, it is recommended to use Graph API and FQL to develop applications instead[22]. All responses from Facebook queries are JSON objects. For example, if I query "http://graph.facebook.com/jingxian.mao" on Facebook, the response will be in the format shown below: In this result, there are 7 fea-

```
{
    "id":"100001401470913",
    "name":"Jingxian Mao",
    "first_name":"Jingxian",
    "last_name":"Mao",
    "username":"jingxian.mao",
    "gender":"male",
    "locale":"en_US"
}
```

Listing 1: An example of result from Facebook

tures (Facebook ID, profile name, first name, last name, username, gender, and locale). There is no need to request permissions for querying the basic information. To get access to additional data including read and write, developers need to request additional permissions. These additional permissions include[23]:

- User and Friend Permissions
- Extended Permissions
- Open Graph Permissions
- Page Permissions

Visual Friends Income Map requests four additional permissions (user's hometown, user's location, user's friends' hometowns and user's friends' locations) from the logged in user (Figure 3.4). Table 3.1 shows the structure and features of the data set that this application gets from Facebook. Table 3.1 is a

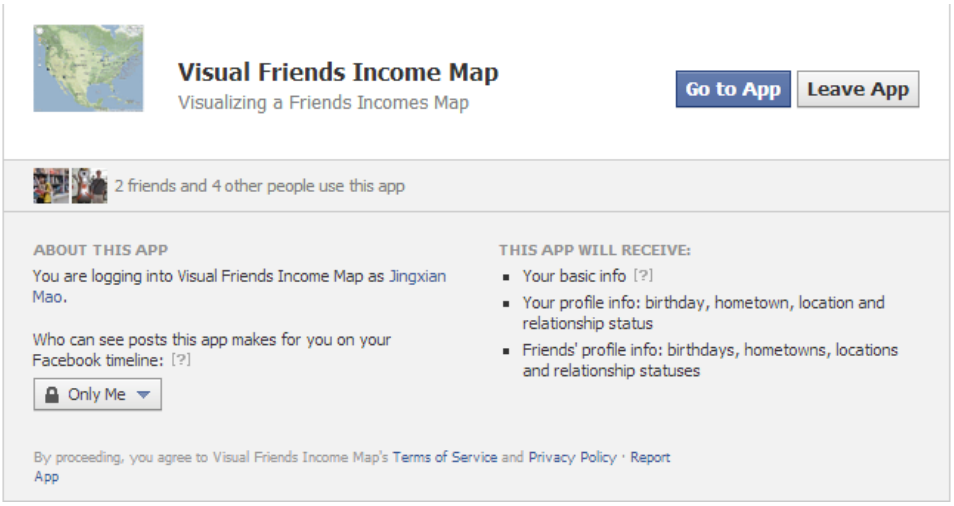

Figure 3.4: Requesting Permissions. If users agree to give out their permissions, this application will start to collect related data from Facebook.

part of the result data set. The size of the data set is equal to the number of a user's friends on Facebook. Every element in the data set has 7 features: (1) an ID is a unique number generated by Facebook, which is the identification of a user; (2) the name is a user's full name; (3) the locations are graph objects, which contain two features - ID and the name of the location; (4) the picture and (5) the link are also both graph objects, where the picture includes a

ID	Name	Location	Picture	Link	Birthday	Relationship
1912748	T P	object	object	object	07/22 /1986	NaN
2323766	N R	object	object	object	05/21	Single
7904730	J T P	NaN	object	object	NaN	NaN
9426585	A S	object	object	object	11/15	NaN
10008736	D H	object	object	object	08/25 /1986	In a relationship
10012913	D M	object	object	object	07/02 /1986	Single
10023247	E M	object	object	object	NaN	Married
10027098	K M	object	object	object	02/28 /1980	Married
10038662	J Y	object	object	object	NaN	NaN
10049381	Z L	object	object	object	10/29 /1984	NaN
10049712	S L	object	object	object	NaN	NaN

Table 3.1: Table of Data set From Facebook

unique ID and an HTTP link that can redirect to a new page, while the link object only provides the HTTP link; (6) the birthday of each user and (7) the relationship status of users. If users have not provided related information, Facebook will return a NaN (shown in Table 3.1, column 2). Taking my personal Facebook data set as an example, I have 369 friends on Facebook. 60 of them have not provided their location information and 20 of them provide an address that is in some other languages, which can be recognized by the application. Therefore, there are approximately 80 records that contain a NaN location object in total.

3.2.2 Census Data

Census data is now commonly used for research, business analysis, marketing, planning, and historical data analysis. By adjusting census samples in polling, analysts and researchers can more efficiently understand problems and solve them. Until now, around 84 counties in the world have made and collected

census data in areas of politics, education, economy and sociology. The Census Bureau of United States[48] is one of the oldest agencies and provides a large historical database that stores all census surveys since 1903. The topics of the surveys cover a variety of fields that provide social norms about the population database. For this work, I downloaded four data sets from the Census Bureau. These four data sets contain the household, family, married couples, and singles average yearly incomes from 2005 to 2010. The difference between these four data sets is that the geographical location level ranges from census tract (city block) level to state level. As shown in Table 3.2 - 3.5, there are six features in each data set:

GEO_id	GEO_label	HMEAN	FMEAN	MMEAN	NMEAN
0400000US01	Alabama	57655	68275	80927	33317
0400000US02	Alaska	82091	93053	106686	53784
0400000US04	Arizona	67436	77127	88273	45232
0400000US05	Arkansas	53253	62497	72578	31538
0400000US06	California	83483	92942	10769	58117
0400000US08	Colorado	75264	89099	100749	47659
0400000US09	Connecticut	94306	112576	130904	53771
0400000US10	Delaware	74703	86238	99994	46928
0400000US11	District of Columbia	91778	116122	173462	72054
0400000US12	Florida	66323	77033	88958	43530
0400000US13	Georgia	66620	76702	91178	42337

Table 3.2: Census Data Table - State Level.

GEO_id	GEO_label	HMEAN	FMEAN	MMEAN	NMEAN
0500000 US01013	Butler County, Alabama	41165	48558	NaN	23740
0500000 US01015	Calhoun County, Alabama	50337	59307	70313	30121
0500000 US01017	Chambers County, Alabama	40393	47119	NaN	24391
0500000 US01019	Cherokee County, Alabama	49201	57457	NaN	25261
0500000 US01021	Chilton County, Alabama	52296	59421	NaN	28947

Table 3.3: Census Data Table - County Level.

GEO_id	GEO_label	HMEAN	FMEAN	MMEAN	NMEAN
1600000US 0100988	Allgood town, Alabama	46372	53946	NaN	24768
1600000US 0101132	Altoona town, Alabama	48715	55826	NaN	28266
1600000US 0101180	Andalusia city, Alabama	54635	63549	NaN	26085
1600000US 0101228	Anderson town, Alabama	28184	34391	NaN	13151
1600000US 0101396	Anniston city, Alabama	42108	43776	NaN	32469

Table 3.4: Census Data Table - City Level.

- GEO_id - A set of unique numbers that represent elements.
- GEO_label - Name of states, counties, cities and tracts.
- HMEAN - Household average yearly incomes mean value.
- FMEAN - Family average yearly incomes mean value.
- MMEAN - Married couples average yearly incomes mean value.
- NMEAN - Singles average yearly incomes mean value.

3.2.3 Geographical Location Data

Since the location data from Facebook are all in text format which cannot be rendered on Google Maps, a conversion from addresses to geographical latitudes and longitudes is needed. Google Maps API provides such a geocoder - Google Geocoding API to convert the addresses into graphical coordinates. However, only 2,500 geolocation requests are allowed by Google geocoding API[28] per day. The limitation will block my application if there are more than 10 users who have used my application within one day. To avoid this limitation, I used the U.S. City and County Web Data API created by U.S. Small Business Administration[55]. This API provides city and county location data with geographical coordinates and a "mashup" of URLs for official city

GEO_id	GEO_label	HMEAN	FMEAN	MMEAN	NMEAN
1400000US01001020100	Census Tract 201,Autauga County, Alabama	91718	108074	NaN	27641
1400000US01001020200	Census Tract 202,Autauga County, Alabama	49125	60543	NaN	29311
1400000US01001020300	Census Tract 203,Autauga County, Alabama	54283	65462	NaN	26361
1400000US01001020400	Census Tract 204,Autauga County, Alabama	65231	73400	NaN	39807
1400000US01001020500	Census Tract 205,Autauga County, Alabama	73316	84535	NaN	47827

Table 3.5: Census Data Table - City Block Level.

and county government web sites. It is RESTful and the output can be in the format of XML and JSON (Listing 2). The output has 14 attributes in one entry. I extracted five of them and created a new data set called "US city and county" (Table 3.6). The US city and county data set stores every city

County_full	City	lat	lng	State
Anchorage Municipality	Anchorage	61.21	-149.9	Alaska
Bethel Census Area	Bethel	60.7922	-161.75	Alaska
Valdez-Cordova Census Area	Cordova	60.54	-145.75	Alaska
Dillingham Census Area	Dillingham	59.0397	-158.458	Alaska
Haines Borough	Haines	59.23	-135.44	Alaska

Table 3.6: US city and county table.

```
{
    "county_name":"Maricopa",
    "description":null,
    "feat_class":"Populated Place",
    "feature_id":"1280",
    "fips_class":"C1",
    "fips_county_cd":"13",
    "full_county_name":"Maricopa County",
    "link_title": null,
    "url":"http://www.ci.wickenburg.az.us/",
    "name":"Wickenburg",
    "primary_latitude":"33.96",
    "primary_longitude":"-112.72",
    "state_abbreviation":"AZ",
    "state_name":"Arizona"
}
```

Listing 2: An example of City and County Web Data API result in JSON format

and county data with the attributes shown in Table 3.6. Now, to geocode the addresses of Facebook users, I just need to query the address in US city and county data set for the geographical coordinates (latitude and longitude).

3.3 Data Analysis

In section 3.1, I introduced my application on Facebook and provided a framework of the application - Visual Friends Income Map. In this section, I introduce and discuss the data analysis methods, predictors that I have used and implemented in my work.

As mentioned in section 3.2.1, there is a group of people who do not provide their location information on Facebook. The goal of this section is how to predict the geographical location of these people. I downloaded a publicly accessible telephone data set that includes the geographical location of about 78 million people who live in US. The data are collected by telephone

directories with geographical latitude and longitude values. Table 3.7 shows the US census geographical location data set. This data set is not up to date,

Surname	Address	City	State	Postcode	Lat	Lng
ABLAH	Monkey Island	Afton	OK	74331	36.822	-94.916
ABBOTT	502 NW D St	Antlers	OK	74523-2059	34.215	-95.626
ABSHIER	NW Of City	Anadarko	OK	73005	35.167	-98.326
ABUCKLE	24512 E Brandy Cir	Afton	OK	74331	36.822	-94.916
ABRAHAM	Shabngra-La Estates	Afton	OK	74331	36.822	-94.916

Table 3.7: US Census Geographical Location Table.

which may cause some errors in finding people's current locations. Furthermore, there are only 78 million records in the data set, which is about 4/15 of the US population. It means that over 222 million people's information are not included in this data set. Therefore, a kernel density estimation method is implemented to compute the population density based on surnames.

Kernel Density Estimation Predictor

Kernel density estimation (KDE) is a non-parametric way to estimate the probability density function of a random variable[57]. It can be used in a data smoothing problem. By applying the KDE, analysts have a smoothed data distribution. Based on this distribution, analysts can make a prediction on a data set with specific attributes.

The efficiency and accuracy of KDE are decided by an estimating function f, which is usually referred to as a "kernel density estimator". A common expression of the kernel density estimator is

$$\hat{f}_h(x) = \frac{1}{nh} \sum_{i=1}^{n} K(\frac{x - x_i}{h}), \tag{3.1}$$

where n is the size of data set; h is the bandwidth; and $K(\frac{x-x_i}{h})$ is the kernel. The kernel is a symmetric non-negative weighing function, which always follows two requirements:

- $\int_{-\infty}^{+\infty} K(u)d(u) = 1$;
- $K(-u) = K(u)$ for all values of u.

As the core of kernel density estimation, eight common kernel functions are widely used: uniform, triangle, Epanechinikov, quartic (biweight), tricube, triweight, Gaussian, and cosine. In this project, both the Epanechinikov and the cosine function can be good kernel functions because both of them provide a peek that falls in the mid-range of $[0, 1]$ and limit the domain of u to be $[-1, 1]$. A mid-range peek smooths the data set by lowering the average slope. The limitation on domain ensures that each friend node affects a limited area on a map. The Epanechinikov function is selected to be the kernel in the thesis. It is

$$K(u) = \frac{3}{4}(1 - u^2), \text{where } |u| \leq 1. \tag{3.2}$$

The data for the kernel density estimation overlay is gathered from US telephone geographical location data set. As shown in table 3.7, each data element has seven attributes: surname, address, city, state, postcode, latitude, and longitude. The responding result is a data set that consists of all records that have the same surname as the application users. The next problem is how to apply the kernel density estimator to this set of data. Considering that the goal of applying the kernel density estimation method is to predict a user's geographical location and draw a heat map on the Google Maps, a good solution is to compute the KDE value of each pixel within the canvas. However, a canvas with a size of 640 * 480 can take up to 10 minutes to scan over all

pixels of this canvas, which is very slow. Another way is to assign a bandwidth (a radius in my approach) to every friend node that occurs within the canvas. It firstly converts the friend nodes' geographical locations to window-based coordinates. Then it sorts the nodes by their coordinates, first in the x direction and then in the y direction. The next step is to compute the KDE value of pixels that are located within the radius of each friend node. The kernel function is

$$K(u) = \frac{3}{4}(1 - u^2), \text{where}, u = \frac{dist_p}{radius}. \tag{3.3}$$

The $dist_p$ in (3.3) is the distance from a pixel within the range of its related friend node to the friend node pixel. The $radius$ shows the range of each friend node on the Google Maps canvas. Theoretically, every pixel that falls in the range of the friend nodes has a KDE value after this approach. In addition, some pixels may fall into a range intersection of multiple nodes. Therefore, a summation of the KDE value for each non-zero pixel is required. The kernel density estimator then is

$$\hat{f}_h(dist_p) = \frac{1}{nh}\sum_{i=1}^{n} K(\frac{3}{4}(1 - (\frac{dist_p}{h})^2)), \text{where}, h = radius. \tag{3.4}$$

Since friend nodes are very sparse, there will be less computations needed in this approach. This approach can be finished in a few seconds.

After the KDE value of each pixel of the map is computed, the next step is to pick a pixel as the person's location. Instead of picking the overall maximum KDE value or the local maximum values of the friend visual map, a random threshold is used for this predictor. To make the algorithm work, the summation of each pixel's KDE value on the map should be normalized to 1. After normalization, the algorithm can be easily implemented: (1) Randomize a threshold that is between 0 and 1. (2) Scan the map row by row and sum up the values of pixels that have been scanned. (3) If the summation is greater

than or equal to the threshold, store this pixel's canvas coordinates; otherwise, keep scanning until the summation reaches the threshold. Once the pixel is found on the canvas, a conversion from canvas coordinates to geographic coordinates - latitude and longitude is implemented. With the geographical coordinates, this application then locates this node at the correct location on the friend visual map.

IP Address Based Predictor

This predictor is created to make the location prediction more accurate. Implementing this predictor needs an IP address database. Geo IP Database (developed by EasyjQuery[18]) has been used in this application. This database can be used with Javascript and PHP. It outputs not only IP address and county but also city block and time zone. When a user runs this application through his or her Facebook account, this predictor automatically looks up the user's IP address in this database and outputs a JSON format result that includes the county, city block, time zone and some other information. After retrieving the addresses of users, this application queries the yearly mean income in the census tract data table to predict users' incomes.

Incomes Predictor

The incomes predictor is a location-based predictor. The core of this predictor is linking and matching the census data with Facebook data. The CensusBureau provides four levels of free resources, which are the state level data set, the county level data set, the city level data set, and the census tract level data set. The accuracy of predictors increases when the level is narrowed down from the state level to the census tract level.

Most Facebook users only provide their location information specific to the city level. Some Facebook users do not provide any private location information on Facebook. Due to these unpredictable situations, linking and matching census data with Facebook information becomes complicated.

As discussed in section 3.2.2, I gathered all levels of data sets from the CensusBureau that include yearly mean incomes and location information. The location information from Facebook is split into county and city, then this application matches the Facebook users' locations with census data in the tables of city level, county level and state level. If there is no match in a lower level, the application then tries to match the Facebook users' location in a higher level data table. It takes the about 0.03 seconds (on average) to link the census data with the Facebook data. The linkages display Facebook users' predicted incomes according to the yearly mean incomes of where they live.

Table 3.2 - 3.5 show that the census data sets from CensusBureau have four types of yearly mean incomes. Instead of only assigning the single yearly incomes value to every person, making a prediction on users' incomes based on other attribute may result in a better prediction. I collected users' age and relationship status information from Facebook. With this information, an age and relationship based predictor is created. This predictor clusters people into four groups based on their age:

- Under 20,
- Over 20 and under 30,
- Over 30 and under 55,
- Over 55.

Then it divides people into four groups based on their relationship status:

- Single,
- Married,
- Family,
- Household.

There could be 16 combinations of these two groups. But only seven combinations are used in the predictor, which are:

- Under 20 and single,
- Over 20 and under 30 and single,
- Over 20 and under 30 and married,
- Over 30 and under 55 and single,
- Over 30 and under 55 and married,
- Over 30 and under 55 and family,
- Over 55 and household.

The reasons for choosing these seven combinations are as follows: (1)People under 20 years old are teenagers and most of them are still single. (2) People over 20 and under 30 can be single, married and in a family. Most people between their 20s and 30s do not have children while married. They probably cannot form a family in this age range. Therefore, if a person's relationship status is "engaged" or "married", this person is assigned into the group "over 20 and under 30 and married". Otherwise, this person is in the other group

that falls in the age range from 20 to 30, over 20 and under 30 and single. (3) People over 30 are probably married and have formed a family so that there are three groups in the age range from 30 to 55, single, married, and in a family. If a person's age is from 30 to 55 and engaged or in a relationship, this person is considered to be in the group of over 30 and under 55 and married. If a person is married, this person has most likely already formed a family. (4) Most people who are over 55 may have a big family so that there is only one combination over 55.

For the people who are in group 1, the predictor assigns the single yearly income value to them. The incomes of people in group 7 are linked to their household yearly incomes. The incomes of other groups are predicted with a linear combination of two yearly incomes. For group 2 and group 4, the household and single yearly incomes are used; for group 3 and group 5, single and married yearly incomes are combined; for group 6, the married and family yearly incomes are applied. The difference between the predictors in group 2 and group 4 is the linear factor (k in equation 3.5).

$$Income = k * value_1 + (1 - k) * value_2 \tag{3.5}$$

where k is the linear factor which is in the range of $(0, 1)$.

3.4 Data Visualization

This section introduces and discusses the data visualization tools that I have implemented.

3.4.1 Overlays on Google Maps

I rendered three overlays on Google Maps: (1) a friend geographical location overlay, (2) a census data overlay, (3) a kernel density estimation overlay. I will introduce each of them in the following sections.

Friend Geographical Location Overlay

There are several ways to render a friend geolocation overlay in Google Maps. Google itself has provided the google.maps.Symbol object class to solve this problem[29]. Developers can customize the symbol's shape, fillcolor, scale, position, stroke, etc. However, the symbol class and marker class that Google provides are not very flexible. D3.js, which is a free online JavaScript library for data analysis and visualization [4], and it introduces a way to deal with layers on Google Maps. D3 is a jQuery-liked library built based on HTML, SVG and CSS. Developers can easily access the HTML document object model(DOM) object by applying the "select" function. D3 also provides other useful methods that help developers to deal with data sets and rendering methods. For example, ".data(d3.entries(data))" is a method that inputs a whole data set. Since D3.js is SVG based library, rendering friend nodes on Google Maps can be easily and efficiently done by D3.js by adding "svg:circle" objects to the target web page. Figure 3.5 shows my Facebook friends (blue

Figure 3.5: Friends Map on Google Maps

circles) on the Google Maps. The location information comes from Facebook

by retrieving my friends' location. Then the application converts the location data into geographical coordinates by querying the table "us_city_county in database "census" and renders friend nodes on Google Maps by adding "svg:circle" object to the Google Maps canvas. A sequence color scheme is used on this layer. As mentioned above, there are several tables with a prefix name "acs" which are from CensusBureau. These tables contain the estimate yearly mean incomes of each state/county/city of the US. When this application gets the responses from Facebook that carries Facebook friends' information, it posts a query request to these tables for estimated yearly income values. Based on income values, people are then divided into 10 groups with different colors. A reason why a sequence color scheme is used is that the yearly income can be any number that is bigger or equal to 0. The sequence color scheme [13] can effectively help people to understand the relations among all nodes. For example, if the color is darker, it means the estimated yearly income is higher.

Census Data Overlay

This overlay shows the estimated yearly mean incomes of each state/county. Plenty of ways have been developed to render such a layer on Google Maps. A common way is to download the shapefile of each state and county and convert them into KML file, and then apply the Google Fusion Table layers. To apply the Fusion Table, developers need to upload the data sets to the Fusion Table, which includes the KML format geometry data, geographic name and census data. Then they apply Fusion Table API[27] to draw out the layer. As shown in Figure 3.6 and Figure 3.7, the value of each state/county is set with the estimated yearly mean incomes. The sequence color scheme is also applied to visualize 10 different groups. The census data overlay is initialized on the

state level. When a user zooms in to a specific zoom level (level 5), the census data overlay would be switched to the county level.

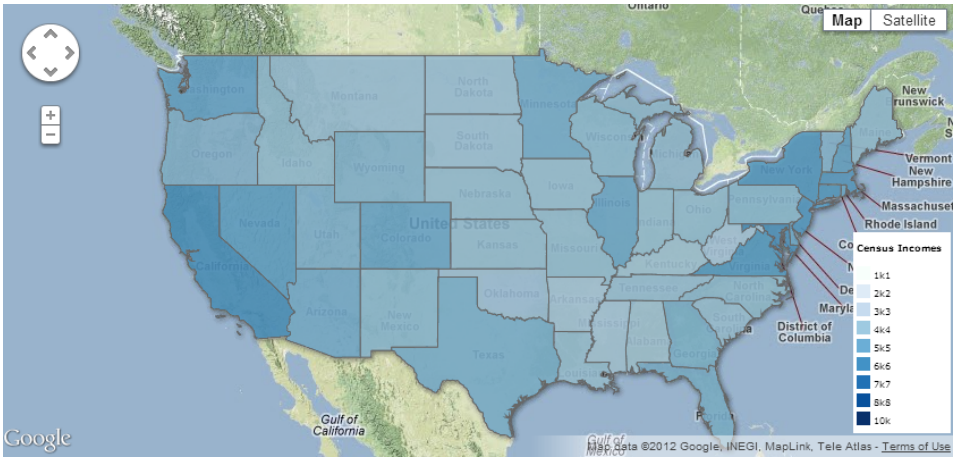

Figure 3.6: Census Data Overlay - State Level: Zoom level from 1 to 4

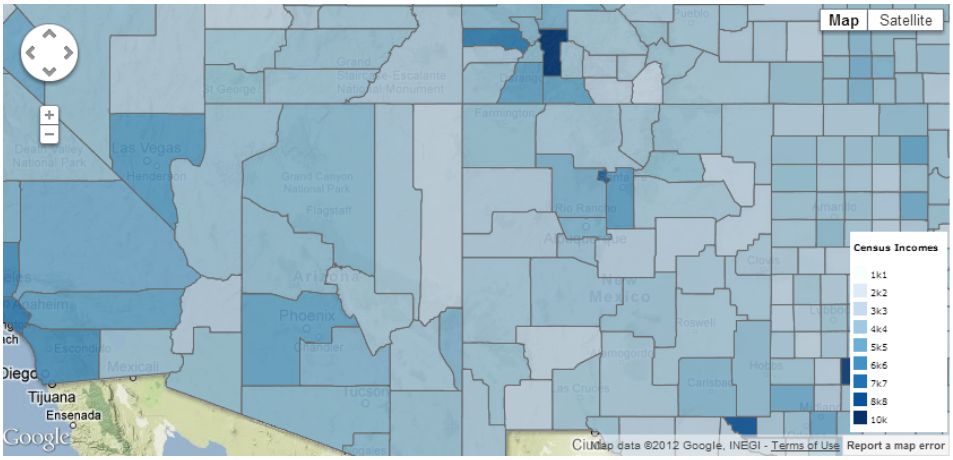

Figure 3.7: Census Data Overlay - County Level: Zoom level over 5

Kernel Density Estimation Overlay

Discussed in Section 3.3.1, the application applies kernel density estimation to compute the KDE value of each pixel within the canvas. With all these KDE values, this overlay is rendered by applying Google Heatmap layers API[30]. This API takes data points that consist of Google Maps LatLng values and

weighted values. The Google Maps LatLng value is converted from canvas coordinates. The weighted value is the KDE value of each pixel. A result of making KDE overlay for my friend "Gray" is shown in Figure 3.8.

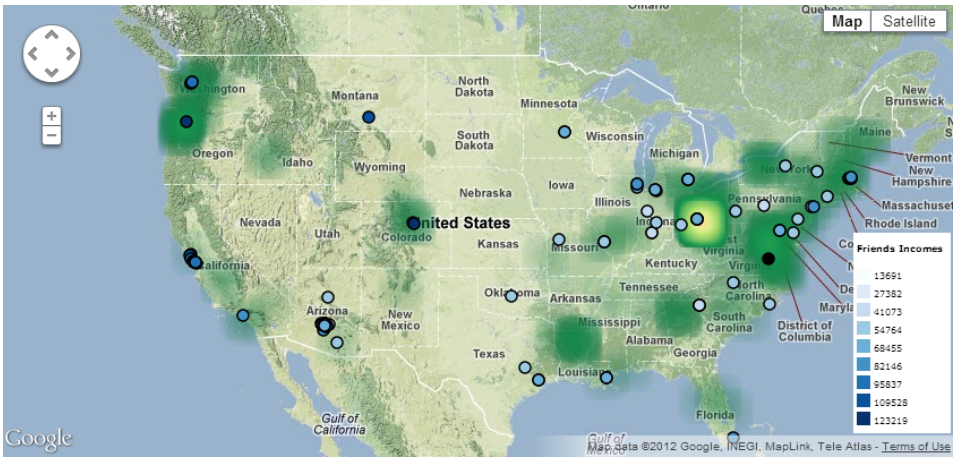

Figure 3.8: Kernel Density Estimation Overlay

3.4.2 Node-Link Network

The Google Maps visualization only displays the geographical location of each person. To explore the relationships between the application user and his/her friends, a node-link network was created (Figure 3.9). This node-link network is based on force-directed algorithm. The term "force-directed" was first used by Fruchterman & Reingold in the 1990 University of Illinois technical report [25]. This algorithm is distinguished from other node-link network algorithms in the way of computing edge length and node position. A node represents a Facebook user and an edge showing the relationships between two users. If there is an edge that connects two nodes, it means that these two people are both friends with each other. When the number of nodes is not too big, there are a lot of algorithms that can make the overall network optimized. When the number of nodes is big enough, how to find the right position for each node to optimize the network becomes a challenge. The force-directed algorithm,

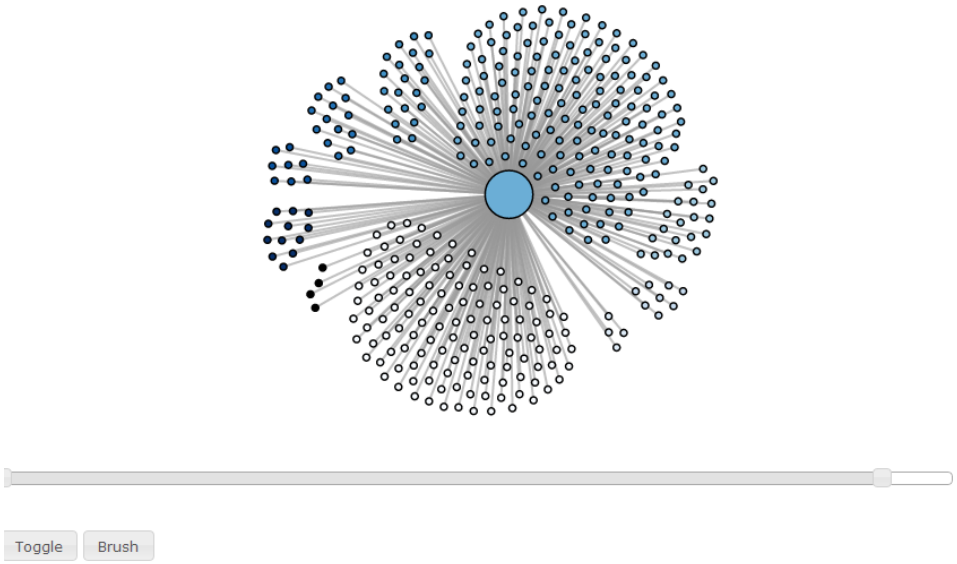

Figure 3.9: Node-Link Network

which is a physical simulation, assigns spring forces among the set of edges and the set of nodes. The spring forces between two nodes pull them together or push them far apart to minimize the summation of all forces in this simulated physical system. I created 10 foci for the node-link network because I clustered the friend nodes into 10 groups based on their estimated yearly mean incomes. The size of each node is its edge degree. The color of each node represents the group it belongs to. A slider bar was created at the bottom of the node-link network. Users can drag the bars to select an incomes range to filter out the nodes. At the same time, the Google Maps is also updated (Figure 3.10). Another interesting interaction of the node-link network is brushing. To apply the brushing method, the "Brush" button shown in Figure 3.9 needs to be activated. Once it is activated, users can brush multiple nodes in the node-link network. The selected nodes will be colored by red circle both in the node-link network and on the Google Maps. By applying this method, users can not only knows their friends' locations, but also their relationship

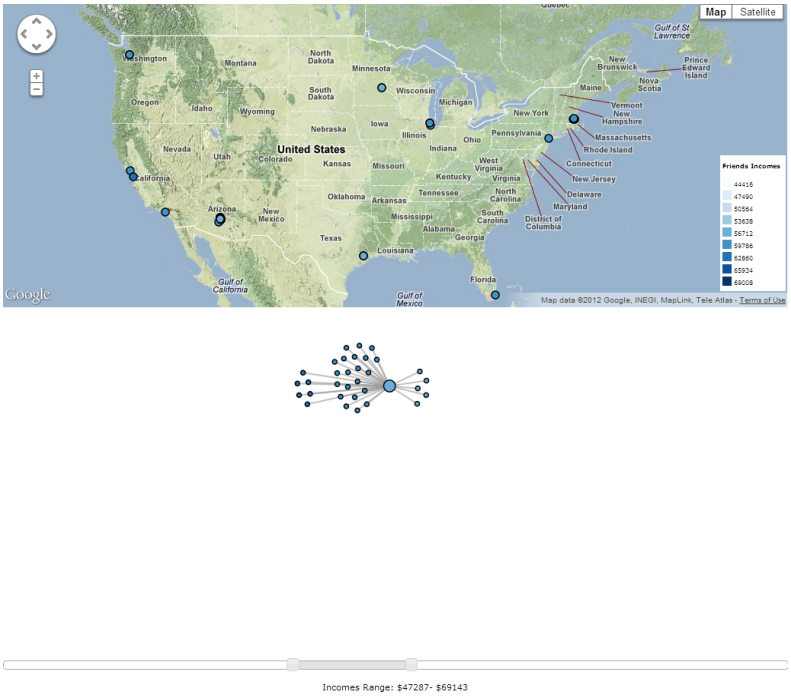

Figure 3.10: Filter Result

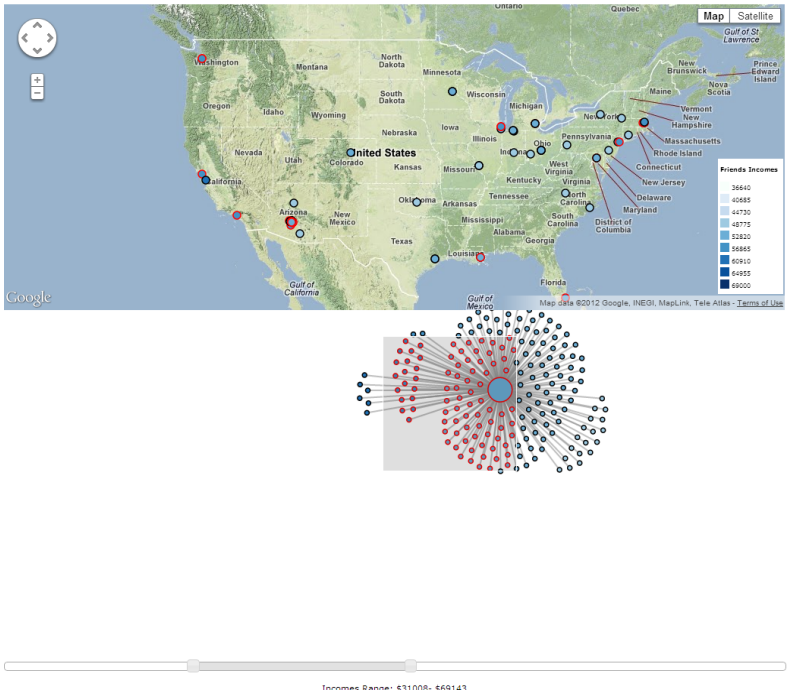

Figure 3.11: Brushing

to them, as well as have a better idea about the groups in which their friends belong to (Figure 3.11).

Chapter 4

RESULTS AND ANALYSIS

In order to explore the results of our proposed multimodal data fusion, I have released an application in Facebook. My current results focus on two Facebook users.

4.1 Case Study 1

The first volunteer is a female international student. To start with this study, let's first take a look at her background and friends network on Facebook. She came to the US for her master program in Arizona State University 3 months ago and created a Facebook account a week later. She has only 39 friends on Facebook and most of them are her classmates or are from the same country, China. The volunteer logged into her account and went to the application through her Facebook. The application then redirected to the result page that consisted of a result region in text format, a friends Google Map, a node-link network and an information window. The friends Google Map, plotted about 90% of her friends who have provided their location information on their own Facebook profiles (Figure 4.1). The other 10% of her friends with location information on Facebook were dismissed from the map because they were out of the US. The results show that almost all her friends are in the area surrounding Tempe, Arizona with only one friend outside the state living in Georgia, USA. Since not all her friends have put their location information on Facebook, the next step is to predict these users' location. The white nodes in Figure 4.2 represent the users who have not shared their location information. To predict these users' incomes, she selected each node and clicked the node's name in the information window. If there are records found in the US census geographical location data table, a KDE predictor is then applied and returns

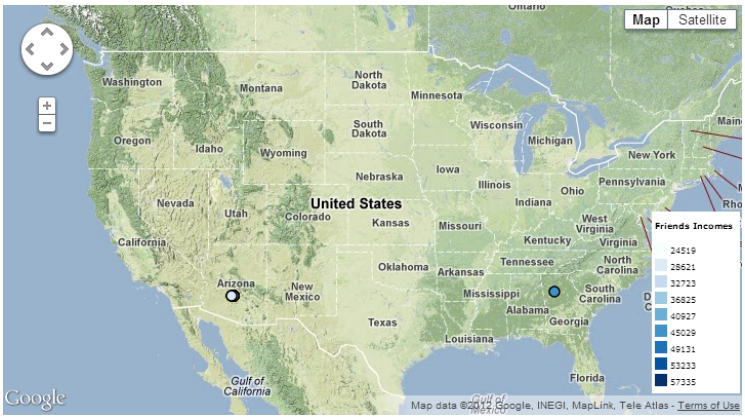

Figure 4.1: Friends Google Map before the location prediction

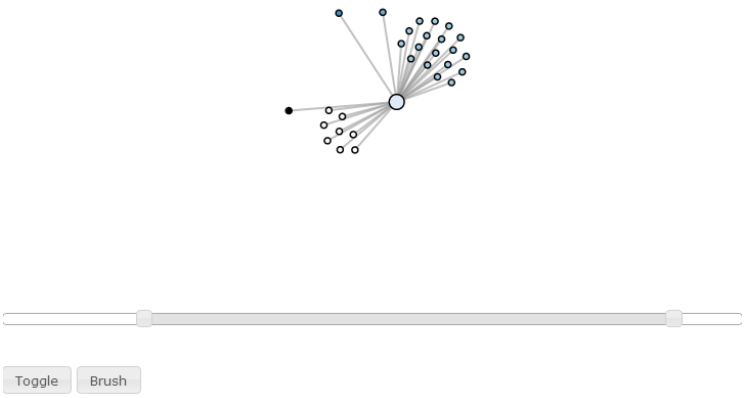

Figure 4.2: Friends Network before the location prediction

the predicted location. The result is shown in Figure 4.3. After the location predictor assigns the predicted location to the selected user, this application automatically runs the incomes predictor on the user. With the predicted income and location information, the user is rendered on the friends Google Map and their friends network is updated as well(Figure 4.4).

Figure 4.3 renders all her friends on the friends Google Map after the application predicts all her in-located friends' location. In Figure 4.3, her

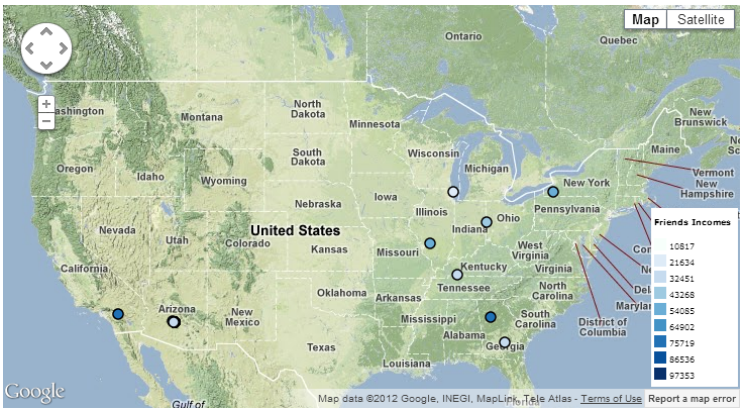

Figure 4.3: Friends Google Map after prediction

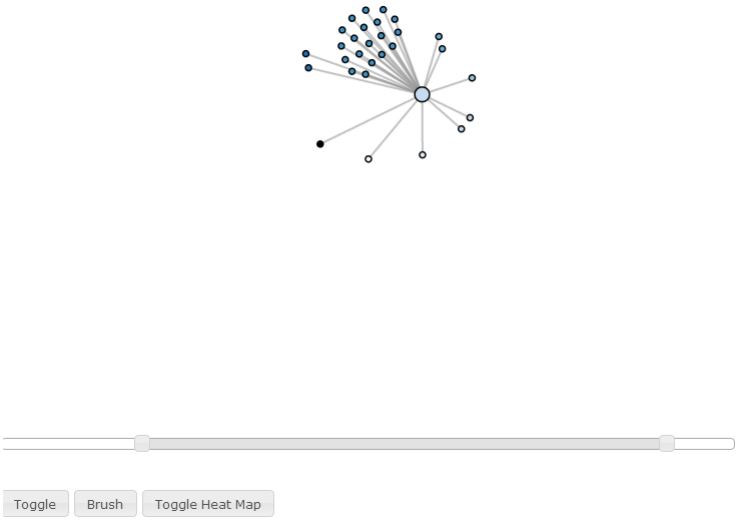

Figure 4.4: Friends Network after prediction

friends are located all around the US. Since her friends are all in, or near, Tempe area in Arizona State except for the one in Georgia, the location predictions are not accurate. How to avoid the prediction mistakes is a big issue in regards to the location predictors. With the prior knowledge that all her friends live in or near Tempe area, I zoomed and panned the map to let the canvas only cover a specific area (state of Arizona). Discussed in Section 3.3.1, the KDE values are computed based on the canvas pixels. Thus zooming and panning the map to a specific area makes the KDE predictor only compute the

KDE value of locations that are displayed in this canvas. Figure 4.5 and Figure 4.6 have explained the differences. In Figure 4.5, the canvas covers the whole US map. Therefore, the KDE predictor computed every location that is in the US. There are several "hot spots" on the map which show the places that have a high density of population containing the surname "He". As shown in Figure 4.5, these "hot spots" areas are near San Francisco, Los Angeles, Chicago and New York City. As I know, these cities have big Chinatowns. Therefore, the results of the KDE prediction over the whole US would most likely fall in or near these "hot spots". (Shown in Figure 4.5 and 4.7). The result in Figure 4.6 shows a good prediction of the KDE predictor. The user "He" from China, is a close friend to the application user. Since He is currently living in Arizona State, I zoomed and panned the map to the state of Arizona and applied the KDE predictor again. Now S He is located in Chandler which is very close to Tempe. Shown in Figure 4.6, in the state of Arizona, most people whose surname is "He" live in the metropolitan Phoenix area or Tucson. Therefore, the person He has a high probability to live in these two places. As the result stands, the KDE predictor guesses He's location is in Chandler which is much more accurate than the previous predictions over the entire US. To make sure that the zooming and panning method is correct, this application runs the KDE predictor on another person whose last name is "Alex". The results are shown in Figure 4.8 and 4.9. Without zooming and panning to a specific area, this person is predicted to overall the country and the results change a lot each time. With the zooming and panning method applied, this person is located in Phoenix most of time, which is quite accurate.

The application outputs the prediction of her yearly income which is around 20,000 dollars. The application also shows that she ranks 0% compared

Figure 4.5: KDE Prediction on All US States

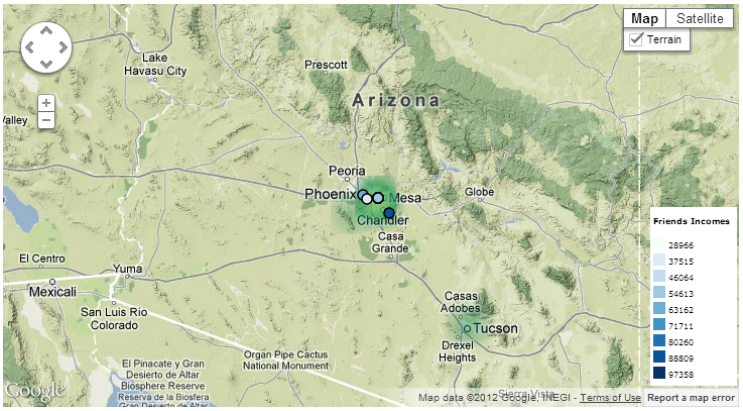

Figure 4.6: KDE prediction on Specific States

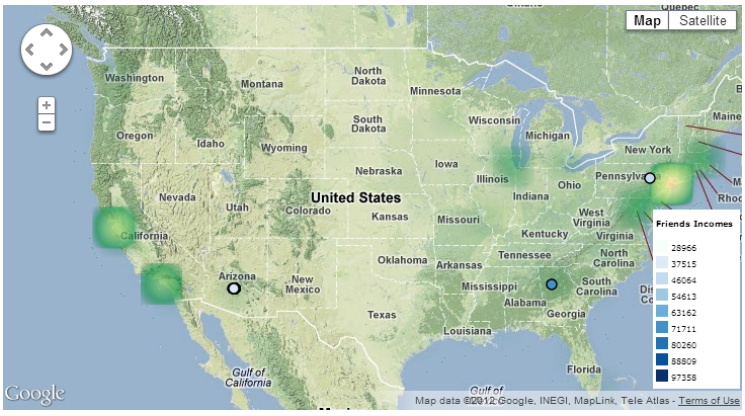

Figure 4.7: KDE Prediction on All US States

to her friends and only 3.39% overall the US population by using the equation:

$$ranking = \frac{MaximumIncome - MinimumIncome}{User'sIncome - MinimumIncome} \tag{4.1}$$

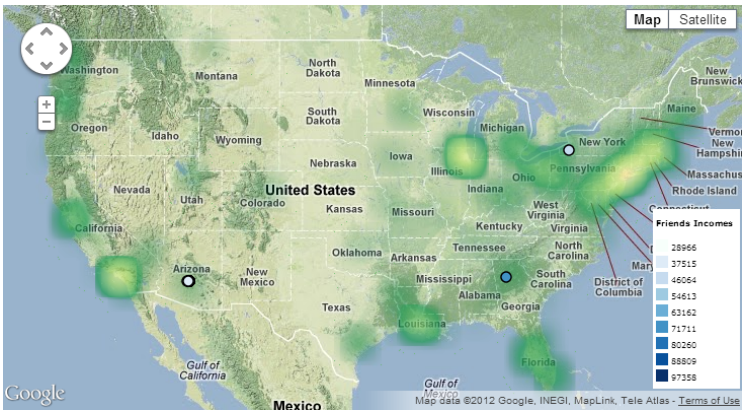

Figure 4.8: KDE Prediction on All US States

Figure 4.9: KDE prediction on Specific States

Figure 4.10: Information Window

Figure 4.11: Information Window

This means that she earns the least among her friends and only more than 3.39% people in US. Figure 4.10 shows that even though the application user and her friends live in the same place, their incomes are different. This is because the IP based location predictor is applied on the user. When a user runs the application, it gets the user's IP address and converts it into geographical coordinates. Since the IP address returns the address that narrows to a city block level, the application looks for the matches of location in the census tract table instead of the city level table. This causes the application user's income to be different from her friends' while they are in the same place.

Not only does the geographical location information have an influence on incomes prediction, but also other users' properties. As mentioned in section 4.3, an age and relationship based predictor is applied in this application. As shown in Figure 4.11, the user "J Gigantino" earns about 90,000 each year while others' incomes are around 50,000 dollars per year. Because Gigantino is over 30 and married, this application applies the linear equation $Income = k * value_1 + (1 - k) * value_2$ to calculate his income. The $value_1$

is the married yearly income and the $value_2$ is the single yearly value. k is a random value that falls in $(0, 1)$. However, other people shown in Figure 4.11 are under 30 and single, so the $value_1$ and the $value_2$ are household yearly incomes and single yearly incomes from census table. The differences in the $value_1$ and the $value_2$ result in different income predictions. The incomes prediction will be discussed more in following case studies.

4.2 Case Study 2

The second volunteer is also an international student from China. I registered my Facebook account 3 years ago when I came to US the first time. After living in US almost 3 years, I have 385 connections on Facebook. These connections are distributed across the US (Figure 4.12). Among all my friends, about 30 percent are my former classmates or schoolmates from China, about 40 percent are my classmates in the US, and the rests are my other friends in US. Over 70 percent of my Facebook friends have graduated and work as full-time employees. Shown in Figure 4.13 and Figure 4.14, the incomes range of

Figure 4.12: Friends Google Map of "Jingxian Mao"

Mao's friends' income range is from 50,000 to 88,000. Now, let's take a look at the people's incomes shown in the information window. A Williams, who is 25 years old and single, is now sharing her house with other roommates in

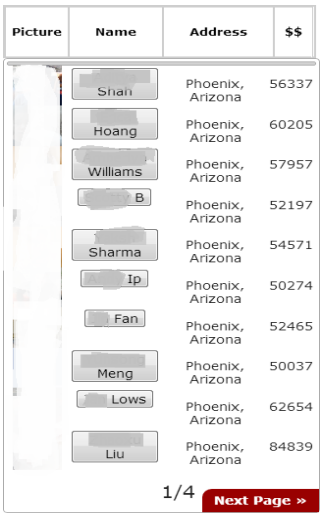

Figure 4.13: Information Window of People in Phoenix

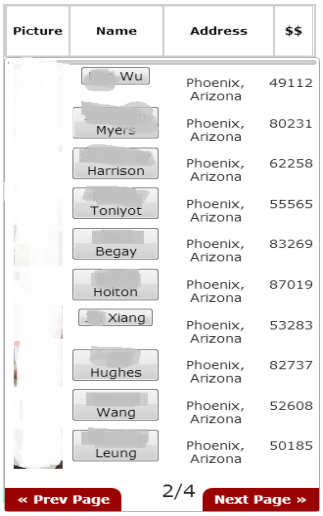

Figure 4.14: Information Window of People in Phoenix

Phoenix. She falls into the group of over 20 and under 30 and single. Therefore, her income is predicted to be 57957 dollars. Her actual income is about 50000 dollars. The error of this prediction is 0.1373. Z Liu, 25 years-old, just married his wife this year. The prediction of his incomes is 84839 dollars according to the age and relationship predictor of the group over 20 and under 30 and married. His actual personal income is about 90000 dollars per year. The error is 0.0573. More examples are listed in the table below (Table 4.1). All the

actual incomes are gathered from the people themselves. They only provide a number which is closest to their incomes. As shown in Table 4.1, the age

Name	Age	Relationship	Location	Predicted	Actual	Error
B H	30	Married	PHX,AZ	87019	85000	0.0237
R T	27	Unknown	PHX,AZ	55565	65000	0.1698
J M	30	Married	PHX,AZ	80231	80000	0.0028
C G	24	Single	PHX,AZ	54270	30000	0.809
Q L	29	Married	Bellevue,WA	114545	120000	0.0454
M C	25	Single	Bellevue,WA	98944	100000	0.0105
M C	29	Unknown	SEA,WA	69273	100000	0.4435
Z L	28	Unknown	Santa Clara,CA	88910	110000	0.2372
N R	Unknown	Single	Waltham,MA	62099	60000	0.0340
C L	Unknown	Married	Richmond,VA	106723	70000	0.5246
L J	Unknown	Married	Gilbert,AZ	75167	75000	0.0022
J C	Unknown	Couple	Chandler,AZ	69831	80000	0.1271
M M	32	Married	Tempe,AZ	80705	95000	0.1505
D H	26	Couple	Tempe,AZ	56576	70000	0.1918

Table 4.1: People's Incomes Table.

range is from 20 to 30 and the relationship status includes "single", "married", "in a relationship", and "unknown". The income predictor is a combination of three properties, which are age, relationship status, and location. Table 4.1 shows that the people all have different income predictions. This is because the incomes predictor sets different values to the values of $value_1$ and the $value_2$ in equation $Income = k * value_1 + (1 - k) * value_2$ and initializes different k in each prediction. The $value_1$ and the $value_2$ are set according to which group is the user in. For example, B H is in the group of over 30 and under 55 and married. Then the $value_1$ is equal to the married yearly mean incomes of Phoenix and the $value_2$ is equal to the family yearly mean incomes of Phoenix. With a random value k, the person B H's income is predicted by applying the equation 3.5. Another example is M C. She is 25 years old and lives in Bellevue. The incomes predictor matches her information with census data and get the household yearly mean incomes and single yearly mean incomes of Bellevue. By setting these two values to $value_1$ and $value_2$, her

income is then predicted as 98944 dollars which is quite close to her actual income 100000 dollars per year.

The results above have shown the incomes prediction of the user's friends'. Next I will discuss the prediction on the user himself(herself). For a user of this application, no location information is needed. The IP based location predictor tracks the user's IP address and looks for its associated geographical location through an IP database. I, Jingxian Mao, am a student of Arizona State University. I am now living in an apartment close to campus. My predicted income is 32138 dollars per year, which is different from other predictions of my friends. It is because their incomes are predicted on a city-level, while my income is predicted on a city block level. The students living in this area usually take part-time jobs. The single average income of these people is around 25000 dollars per year. Comparing to the city's (Tempe)single yearly mean incomes, 40000 dollars, the prediction based on the city block is much more accurate. I have not taken any research assistant or teaching assistant position, so my actual income is 0. However, as I know, the average incomes for a research assistant and a teaching assistant is about 20000 dollars. Therefore if a research assistant or teaching assistant student runs my application, his or her predicted incomes would be between around 25000 dollars, which is quite accurate to the accurate income. But if the IP address based predictor is not applied, the yearly incomes of people living in Tempe is over 45000 dollars. This prediction is much worse than the one with an IP address based predictor. Unfortunately, this prediction only works for the person who is using this application, since this application cannot find other machines' IP addresses if they are not connected to the server.

Chapter 5

CONCLUSION AND FUTURE WORK DISCUSSION

This thesis has shown the development of a methodology to study the data in online social networks and explore missing information with multiple predictors. To test our methodology, I have developed an application, "Visual Friends Income Map", which links census data tables with Facebook data. Through this application, Facebook users can explore the estimated incomes of their friends and themselves. Multiple predictors have been used in this application for geographical location and income prediction. My work applied several predictors which include: age and relationship based predictors, KDE geographical location predictor, and IP address based predictor. By combining these predictors together, a multimodal data fusion is then created as a means of extrapolating missing interesting information in Facebook. The location based income predictor provides a first pass approach to predict a person's income based on their geographical location information. From this result other predictors (i.e., age, gender, relationship, surname) are used to refine the results. As the results show, this application is quite accurate when a Facebook user has provided as much information as they can in Facebook, such as age, relationship status, and current location. For those users who provided no information on their Facebook profile, the KDE geographical location predictor is then applied and may lead to an inaccurate result both on their location and income. However, if the application users have some prior knowledge about where their friends actually are, the accuracy of KDE geographical location predictor can be improved. The IP address based predictor lowers the location level to city blocks. Since incomes at the block level are typically more homogeneous, this approach makes the prediction more specific to small area

of a city and results in a higher accuracy on the income prediction. Lastly, the age and relationship based income predictor divides people into different groups to match the data in the census data table.

This application also provides a node-link network and an information window for users to interact with. The node-link network is rendered by the force-directed layout. Friends of the application user are clustered into 10 groups according to their predicted incomes. Brushing and filtering are available for the node-link network. If users are interested in their friend's basic information through Facebook, they can move the mouse onto the matching nodes and the information of the nodes will be displayed in the information window. In this window, a clickable profile picture is available to redirect to this person's Facebook profile page. For those people who do not put their location information on Facebook, a clickable button is created to apply the KDE geographical location predictor. Users can click on this button and the KDE geographical location predictor will be applied. Both the friends Google Map and the node-link network will be updated at the same time after the prediction.

As discussed in this thesis, the income prediction on people who are not students is much more accurate than students. How to solve this problem will be a big concern in the future work. One means to solve this problem is to collect users' education history information from Facebook. If these users are student, then they most likely live near campus or on campus. Therefore, a new location predictor can be created specific for the students. This predictor finds out the campus' location and what city block it is located in. Based on the city block census, a prediction on students can then be applied. As mentioned in case study 1, how the KDE geographical location predictor is applied can

make a huge difference on the location prediction. The best way in which to locate a user's approximate location is a key point to make the KDE location predictor more accurate. One way is to combine the IP address predictor with the KDE predictor. By tracking and restoring the user's friends' IP address, the application can better approximate where the user's friends' locations. However, if none of the user's friends have used this application before, it is not able to track their IP address. Thus, another big step of this application is how to collect and analyze other information such as education history, likes, shares, posts, and check-in statuses to predict this person's location. Last but not the least, how to measure the accuracy of predictions is the most important issue, as well as the most difficult problem. Without an accurate measuring of the prediction accuracy, it is really hard to define if the prediction application is suitable. There are several ways to measure the accuracy of prediction. This is left for future work.

REFERENCE

[1] Sixdegrees.com, 1997. http://sixdegrees.com/.

[2] L. A. Adamic. The Small World Web. pages 443–452. Springer, 1999.

[3] L. Backstrom, E. Sun, and C. Marlow. Find me if you can: Improving geographical prediction with social and spatial proximity. In Proceedings of the 19th international conference on World wide web, WWW '10, pages 61–70, New York, NY, USA, 2010. ACM.

[4] M. Bostock, V. Ogievetsky, and J. Heer. D3: Data-Driven Documents. IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis), 2011

[5] D. M. Boyd and N. Ellison. Social network sites: Definition, history, and scholarship. Journal of Computer-Mediated Communication, 13(1), 2007.

[6] C. Bureau. Census bureau, 2012. http://www.census.gov/.

[7] J. Carrière and R. Kazman. Interacting with huge hierarchies: Beyond cone trees. In Proc. IEEE Information Visualization '95, IEEE Computer Press, Los Alamitos, CA, pages 74–81. IEEE, 1995.

[8] J. Caverlee and S. Webb. A Large-Scale Study of MySpace: Observations and Implications for Online Social Networks. In Proceedings from the 2nd International Conference on Weblogs and Social Media (AAAI), 2008.

[9] CBS. Facebook: One social graph to rule them all?, 2010. http://www.cbsnews.com/stories/ 2010/04/21/tech/main6418458.shtml.

[10] Z. Cheng, J. Caverlee, and K. Lee. You are where you tweet: A content-based approach to geo-locating twitter users. In Proceedings of the 19th ACM international conference on Information and knowledge management, CIKM '10, pages 759–768, New York, NY, USA, 2010. ACM.

[11] E. Cho, S. A. Myers, and J. Leskovec. Friendship and mobility: User movement in location-based social networks. In Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '11, pages 1082–1090, New York, NY, USA, 2011. ACM.

[12] CliffsNotes. Social groups, 2011. http://www.cliffsnotes.com/.

[13] Colorbrewer. Colorbrewer, 2012. http://colorbrewer2.org/.

[14] R. Conrad. Classmates.com, 1995. http://www.classmates.com/.

[15] P. DeScioli, R. Kurzban, E. N. Koch, and D. Liben-Nowell. Best friends. Perspectives on Psychological Science, 6(1):6–8, 2011.

[16] P. Eades. Drawing Free Trees. International Institute for Advanced Study of Social Information Science, Fujitsu Limited, 1991.

[17] P. A. Eades. A heuristic for graph drawing. In Congressus Numerantium, volume 42, pages 149–160, 1984.

[18] EasyjQuery. The Geo IP database, 2012. http://www.easyjquery.com/.

[19] B. Eriksson, P. Barford, J. Sommers, and R. Nowak. A learning-based approach for IP geolocation. In Proceedings of the 11th international conference on Passive and active measurement, PAM'10, pages 171–180, Berlin, Heidelberg, 2010. Springer-Verlag.

[20] Facebook. Facebook FQL, 2012. https://developers.facebook.com/docs/reference/fql/.

[21] Facebook. Facebook graph API, 2012. https://developers.facebook.com/docs/ reference/api/.

[22] Facebook. Facebook old REST API, 2012. https://developers.facebook.com/docs/reference/rest/.

[23] Facebook. Facebook permissions reference, 2012. https://developers.facebook.com/docs/ authentication/permissions/.

[24] L. Freeman. Visualizing social networks. Journal of Social Structure, 2000(Volume 1), 2000.

[25] T. M. J. Fruchterman and E. M. Reingold. Graph drawing by force-directed placement. Softw., Pract. Exper., 21(11):1129–1164, 1991.

[26] P. Gill, Y. Ganjali, B. Wong, and D. Lie. Dude, where's that IP?: circumventing measurement-based IP geolocation. In Proceedings of the 19th USENIX conference on Security, USENIX Security'10, pages 16–16, Berkeley, CA, USA, 2010. USENIX Association.

[27] Google. The FUSION TABLE API. https://developers.google.com/maps/doc-umentation/javascript/layers.

[28] Google. The Google Geocoding API. https://developers.google.com/maps/documentation/geocoding/.

[29] Google. Google Javascript API. https://developers.google.com/maps/documentation/javascript/.

[30] Google. The Layers API. https://developers.google.com/maps/ documentation/javascript/layers.

[31] M. S. Granovetter. The Strength of Weak Ties. American Journal of Sociology, 78(6):1360–1380, 1973.

[32] U. Gretzel. Social network analysis: Introduction and resources, 2001. http://lrs.ed.uiuc.edu/tse-portal/analysis/social-network-analysis/.

[33] B. Gueye, S. Uhlig, and S. Fdida. Investigating the imprecision of IP block-based geolocation. In Proceedings of the 8th international conference on Passive and active network measurement, PAM'07, pages 237–240, Berlin, Heidelberg, 2007. Springer-Verlag.

[34] J. Heer and D. Boyd. Vizster: Visualizing online social networks. In IEEE Information Visualization (InfoVis), INFOVIS '05, pages 32–39, Washington, DC, USA, 2005. IEEE Computer Society.

[35] N. Henry and J.D. Fekete. Matlink: enhanced matrix visualization for analyzing social networks. In Proceedings of the 11th IFIP TC 13 international conference on Human-computer interaction - Volume Part II, INTERACT'07, pages 288–302, Berlin, Heidelberg, 2007. Springer-Verlag.

[36] N. Henry, J.D. Fekete, and M. J. McGuffin. Nodetrix: A hybrid visualization of social networks. IEEE Transactions on Visualization and Computer Graphics, 13(6):1302–1309, 2007.

[37] I. Herman, G. Melancon, and M. S. Marshall. Graph visualization and navigation in information visualization: A survey. IEEE Transactions on Visualization and Computer Graphics, 6(1):24–43, 2000.

[38] B. Johnson and B. Shneiderman. Tree-maps: a space-filling approach to the visualization of hierarchical information structures. In Visualization, 1991. Visualization '91, Proceedings., IEEE Conference on, pages 284 –291, oct 1991.

[39] E. Katz-Bassett, J. P. John, A. Krishnamurthy, D. Wetherall, T. Anderson, and Y. Chawathe. Towards IP geolocation using delay and topology measurements. In Proceedings of the 6th ACM SIGCOMM conference on Internet measurement, IMC '06, pages 71–84, New York, NY,USA,2006. ACM.

[40] R. Keller, C. M. Eckert, and P. J. Clarkson. Matrices or node-link diagrams: which visual representation is better for visualising connectivity models? Information Visualization, 5(1):62–76, Mar. 2006.

[41] E. O. Laumann and L. Guttman. The relative associational contiguity of occupations in an urban setting. American Sociological Review.31, pages 169–178, 1966.

[42] J. Levine. Joint-space analysis of "pick-any" data: Analysis of choices from an unconstrained set of alternatives. Psychometrika, 44:85–92, 1979. 10.1007/BF02293787.

[43] D. Liben-Nowell. Wayfinding in social networks. In G. Cormode and M. Thottan, editors, Algorithms for Next Generation Networks, Computer Communications and Networks, pages 435–456. Springer London, 2010.

[44] D. Liben-Nowell, J. Novak, R. Kumar, P. Raghavan, and A. Tomkins. Geographic routing in social networks. Proceedings of the National Academy of Sciences of the United States of America, 102(33):11623–11628, 2005.

[45] S. Milgram. The small world problem. Psychology Today, 61:60–67, 1967.

[46] P. Mutton. Inferring and visualizing social networks on internet relay chat. In Proceedings of the Information Visualisation, Eighth Interna-

tional Conference, IV '04, pages 35–43, Washington, DC, USA, 2004. IEEE Computer Society.

[47] M. E. J. Newman, D. J. Watts, and S. H. Strogatz. Random graph models of social networks. Proc. Natl. Acad. Sci. USA, 99:2566–2572.

[48] U.S. Department of Commerce. Census bureau, 2012. http://www.census.gov/.

[49] J.-P. Onnela, S. Arbesman, A.-L. Barabási, and N. A. Christakis. Geographic constraints on social network groups. CoRR, abs/1011.4859, 2010.

[50] Z. Papacharissi. The virtual geographies of social networks:A comparative analysis of facebook, linkedin and asmallworld. New Media and Society, 11(1-2):199–220, February/March 2009.

[51] I. Poese, S. Uhlig, M. A. Kaafar, B. Donnet, and B. Gueye.IP geolocation databases: Unreliable? SIGCOMM Comput. Commun. Rev., 41(2):53–56, Apr. 2011.

[52] C. Proctor. Informal social systems. Turrialba, pages 73–78, 1953.

[53] E. M. Reingold, John, and S. Tilford. Tidier drawing of trees. IEEE Trans. Software Eng, 1981.

[54] G. G. Robertson, J. D. Mackinlay, and S. K. Card. Cone trees: Animated 3d visualizations of hierarchical information. In Proceedings of the SIGCHI conference on Human factors in computing systems: Reaching through technology, CHI '91, pages 189–194, New York, NY, USA, 1991. ACM.

[55] SBA. U.S. small business administration. http://www.sba.gov/about-sba.

[56] F. Schneider, A. Feldmann, B. Krishnamurthy, and W. Willinger. Understanding online social network usage from a network perspective. In Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference, IMC '09, pages 35–48, New York, NY, USA, 2009. ACM.

[57] B. W. Silverman. Density estimation for statistics and data analysis / B.W. Silverman. Chapman and Hall, London ; New York :, 1986.

[58] Y. Takhteyev, A. Gruzd, and B. Wellman. Geography of twitter networks. Social Networks, 34(1):73 – 81, 2012. Capturing Context: Integrating Spatial and Social Network Analyses.

[59] USAtoday. Facebook tops 1 billion users, 2012. http://www.usatoday.com/story/tech/2012/10/04/facebook-tops-1-billion-users/1612613/.

[60] F. Viegas and J. Donath. Social network visualization: Can we go beyond the graph? In Proceedings of the Computer Supported Collaborative Work Conference CSCW'04, Workshop on Social Networks, Chicago, U.S.A., 2004.

[61] Y. Wang, D. Burgener, M. Flores, A. Kuzmanovic, and C. Huang. Towards street-level client-independent IP geolocation. In Proceedings of the 8th USENIX conference on Networked systems design and implementation, NSDI'11, pages 27–27, Berkeley, CA, USA, 2011. USENIX Association.

[62] S. Wasserman and K. Faust. Social Network Analysis. Cambridge: Cambridge University Press, 1994.

[63] S. Yardi and D. Boyd. Tweeting from the town square: Measuring geographic local networks. In W. W. Cohen and S. Gosling, editors, ICWSM. The AAAI Press, 2010.